Construction of binary matrices for near-optimal compressed sensing

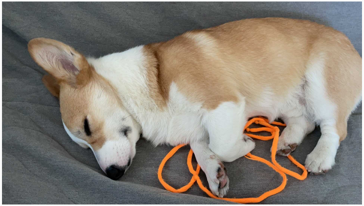
Ivan Lau Jonathan Jedwab

Department of Mathematics, Simon Fraser University

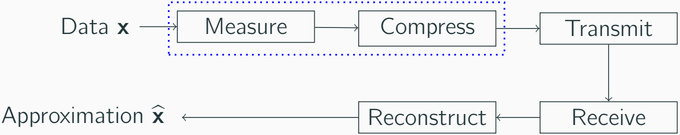
IEEE International Symposium on Information Theory 2021

Motivation for compressed sensing

Original image $\mathbf{x}$: all wavelets



Approximation $\widehat{\mathbf{x}}$: only large-coefficient wavelets



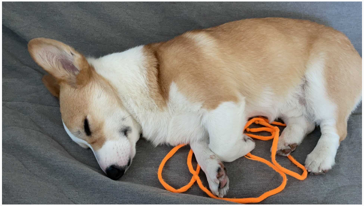
Conventional paradigm for data acquisition:

1. Measure full data (take picture with many pixels)
2. Compress (discard the small coefficients)

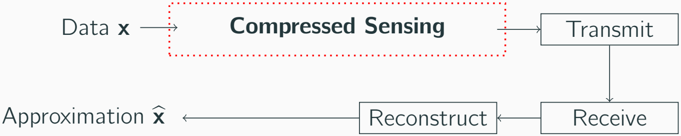
Wasteful: can we measure only the significant part?

Motivation for compressed sensing

Original image $\mathbf{x}$: all wavelets



Approximation $\widehat{\mathbf{x}}$: only large-coefficient wavelets



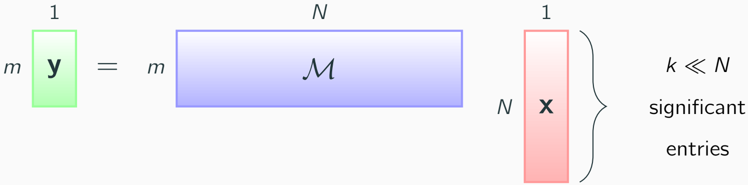
Compressed sensing paradigm for data acquisition:

1. & 2. Directly acquire compressed data

Compress, e.g. discarding the insignificant coefficients

Wasteful: can we measure only the significant part?

Compressed sensing: formal setup



- Wish to recover $\mathbf{x} \in \mathbb{R}^N$ fully from $m \ll N$ non-adaptive linear measurements, i.e. $\mathcal{M}\mathbf{x} = \mathbf{y} \in \mathbb{R}^m$
- Impossible in general: underdetermined system
- $\mathbf{x}$ has $k \ll N$ nonzero entries: exact recovery is possible
- Otherwise, give an approximation $\widehat{\mathbf{x}}$ to $\mathbf{x}$ containing the $k \ll N$ significant entries

Questions:

1. Good measurement matrix $\mathcal{M}$?
2. Recovery algorithm (how to approximate $\mathbf{x}$ using $\mathbf{y}$)?

Efficient compressed sensing schemes

1. Measurement matrix $\mathcal{M}$? 2. Recovery algorithm?

Properties of a good scheme:

(P1) few measurements, ideally $m = O(k \, \mathrm{polylog} N)$

(P2) fast recovery algorithm, ideally $O(k \, \mathrm{polylog} N)$

(P3) few random bits to construct $\mathcal{M}$, ideally $o(N)$

(P4) $\widehat{\mathbf{x}}$ approximates $\mathbf{x}$ accurately via an "ℓ_p/ℓ_q" error guarantee:

$$\|\mathbf{x} - \widehat{\mathbf{x}}\|_p \leq C k^{1/p - 1/q} \min_{k\text{-sparse } \mathbf{x}_k} \|\mathbf{x} - \mathbf{x}_k\|_q$$

for some real constants C and $1 \leq q \leq p \leq 2$

Lower bounds for nontrivial schemes by Ba et al. (2010) :

(P4) $\implies$ measurements, runtime $\Omega(k \log(N/k))$

(Non)uniform recovery

Nonuniform recovery: For each $\mathbf{x} \in \mathbb{R}^N$, generate a matrix $\mathcal{M}$ randomly and independently. With high probability, the error guarantee (P4) is satisfied.

Uniform recovery: Generate a matrix $\mathcal{M}$ randomly. With high probability, the error guarantee (P4) is satisfied for all $\mathbf{x} \in \mathbb{R}^N$.

Principal previous schemes

(P1): number of measurements (P2): recovery algorithm runtime

(P3): number of random bits (P4): error guarantee of $\widehat{\mathbf{x}}$

Schemes good across (P1)–(P4) simultaneously?

Lower bounds	$k\log(N/k)$	$k\log(N/k)$	?	ℓ_2/ℓ_2
–	$k\log k \cdot \log N$	$k\log k \cdot \log N$	$\log k \cdot \log(k\log N)$	ℓ_2/ℓ_1

Paper	**(P1)**	**(P2)**	**(P3)**	**(P4)**
Cormode & Muthukrishnan (2006)	$k\log^3 N$	$k\log^3 N$	$\Omega(N)$	ℓ_2/ℓ_2
Gilbert et al. (2012)	$k\log(N/k)$	$k\log^{\geq 2} N$	$\Omega(N)$	ℓ_2/ℓ_2
Nakos & Song (2019)	$k\log(N/k)$	$k\log^2(N/k)$	$\Omega(N)$	ℓ_2/ℓ_2
Scheme 1, Iwen (2014)	$k\log k \cdot \log N$	$k\log k \cdot \log N$	$\Omega(N)$	ℓ_2/ℓ_1
Scheme 2, Iwen (2014)	$k\log^2 N$	$k\log^2 N$	$\log k \cdot \log(k\log N)$	ℓ_2/ℓ_1
Our Result	$k\log k \cdot \log N$	$k\log k \cdot \log N$	$\log k \cdot \log(k\log N)$	ℓ_2/ℓ_1

The complexities are subject to O-factor, unless stated with Ω.

Principal previous schemes

(P1): number of measurements (P2): recovery algorithm runtime

(P3): number of random bits (P4): error guarantee of $\widehat{\mathbf{x}}$

Schemes good across (P1)–(P4) simultaneously?

Lower bounds	$k\log(N/k)$	$k\log(N/k)$	?	ℓ_2/ℓ_2

Paper	**(P1)**	**(P2)**	**(P3)**	**(P4)**
Cormode & Muthukrishnan (2006)	$k\log^3 N$	$k\log^3 N$	$\Omega(N)$	ℓ_2/ℓ_2
Gilbert et al. (2012)	$k\log(N/k)$	$k\log^{\geq 2} N$	$\Omega(N)$	ℓ_2/ℓ_2
Nakos & Song (2019)	$k\log(N/k)$	$k\log^2(N/k)$	$\Omega(N)$	ℓ_2/ℓ_2
Scheme 1, Iwen (2014)	$k\log k\cdot\log N$	$k\log k\cdot\log N$	$\Omega(N)$	ℓ_2/ℓ_1
Scheme 2, Iwen (2014)	$k\log^2 N$	$k\log^2 N$	$\log k\cdot\log(k\log N)$	ℓ_2/ℓ_1

The complexities are subject to O-factor, unless stated with Ω

Our scheme: combining advantages of Iwen's schemes

(P1): number of measurements (P2): recovery algorithm runtime

(P3): number of random bits (P4): error guarantee of $\widehat{\mathbf{x}}$

Schemes good across (P1)–(P4) simultaneously?

Lower bounds	$k\log(N/k)$	$k\log(N/k)$	?	ℓ_2/ℓ_2

Paper	**(P1)**	**(P2)**	**(P3)**	**(P4)**
Cormode & Muthukrishnan (2006)	$k\log^3 N$	$k\log^3 N$	$\Omega(N)$	ℓ_2/ℓ_2
Gilbert et al. (2012)	$k\log(N/k)$	$k\log^{\geq 2} N$	$\Omega(N)$	ℓ_2/ℓ_2
Nakos & Song (2019)	$k\log(N/k)$	$k\log^2(N/k)$	$\Omega(N)$	ℓ_2/ℓ_2
Scheme 1, Iwen (2014)	$k\log k\cdot\log N$	$k\log k\cdot\log N$	$\Omega(N)$	ℓ_2/ℓ_1
Scheme 2, Iwen (2014)	$k\log^2 N$	$k\log^2 N$	$\log k\cdot\log(k\log N)$	ℓ_2/ℓ_1
Our scheme	$k\log k\cdot\log N$	$k\log k\cdot\log N$	$\log k\cdot\log(k\log N)$	ℓ_2/ℓ_1

The complexities are subject to O-factor, unless stated with Ω.

How to combine advantages of Iwen's schemes?

Measurement matrix:

$$\mathcal{M} = \left[\frac{\mathcal{M}_{\mathrm{id}}}{\mathcal{M}_{\mathrm{est}}}\right] \quad \begin{array}{l} \leftarrow \text{identify indices of significant entries} \\ \leftarrow \text{estimate values of entries} \end{array}$$

Algorithm 1 Recovery Algorithm

Input: $\mathcal{M} = \left[\frac{\mathcal{M}_{\mathrm{id}}}{\mathcal{M}_{\mathrm{est}}}\right]$, $\mathbf{y} = \left[\frac{\mathbf{y}_{\mathrm{id}}}{\mathbf{y}_{\mathrm{est}}}\right] = \mathcal{M}\mathbf{x}$, and $k \in [N]$

Output: an approximation $\widehat{\mathbf{x}}$ to $\mathbf{x}$

1: $S = \mathrm{Identify}(\mathbf{y}_{\mathrm{id}})$ ▷ indices of significant entries

2: $\widehat{\mathbf{x}} = \mathrm{Estimate}(\mathcal{M}_{\mathrm{est}}, \mathbf{y}_{\mathrm{est}}, S, k)$ ▷ estimate entries indexed by S

$\boxed{\text{Our scheme: same algorithm, same } \mathcal{M}_{\mathrm{est}}, \text{ improved } \mathcal{M}_{\mathrm{id}}}$

Our identification matrix: subsample from a better binary matrix

Our scheme: same algorithm, same $\mathcal{M}_{\mathrm{est}}$, improved $\mathcal{M}_{\mathrm{id}}$

Iwen's and our $\mathcal{M}_{\mathrm{id}}$ is generated by

(i) randomly subsampling rows of **"incoherent" binary matrix**,

(ii) then taking "columnwise Kronecker product" with the "bit-tester"

Our $\mathcal{M}_{\mathrm{id}}$: subsample rows from a **better** incoherent binary matrix

Incoherent binary matrix

$\{0,1\}^{t \times N}$ is (w, α)-coherent matrix

1. each column contains at least w 1s,
2. each pair of distinct columns has dot product at most α.

Questions:

1. Lower bound on t?
2. Upper bound on t?
3. Construction?

at least two 1s

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

dot product at most 1

$(2, 1)$-coherent matrix with $N = 6$

Our lower bound on the row count

$\{0,1\}^{t\times N}$ is (w,α)-coherent matrix

1. each column contains at least w 1s,
2. each pair of distinct columns has dot product at most α.

1. **Lower bound on t?** 2. Upper bound on t? 3. Construction?

Our lower bound: $t = \Omega(w^2/\alpha)$

Proof idea (using coding theory):

- Bound must apply to the case with **exactly** w 1s.
- Translate into binary constant-weight code: $(t, 2(w-\alpha), w)_2$-code of size N
- Rearrange classical bound by Johnson (1962): $t = \Omega(w^2/\alpha)$

Iwen's upper bound on row count and constructions

$$\boxed{t = \Omega(w^2/\alpha)}$$

1) Scheme 1 (best (P2), fastest recovery algorithm)
   - Randomly generated itself
   - $t = O(w^2/\alpha)$, **order-optimal**!

2) Scheme 2 (best (P3), fewest random bits)
   - **Explicit** construction, based on RIP matrix by DeVore (2007)
   - $t = O(w^2)$

	(w, α)-coherent matrix		Performance of scheme		
Scheme	Row count	Explicit	(P1)	(P2)	(P3)
Iwen's scheme 1	$O(w^2/\alpha)$	✗	good		poor
Iwen's scheme 2	$O(w^2)$	✓	poor		good

Combining the advantages?

Our matrix construction: explicit and order-optimal

Advantage in (w, α)-coherent matrix	Corresponding advantage(s) in scheme
Good row count	few measurements (P1), fast runtime (P2)
Explicit (structured)	few random bits (P3)

Combining the advantages?

	(w, α)-coherent matrix		Performance of scheme		
Scheme	Row count	Explicit	(P1)	(P2)	(P3)
Iwen's scheme 1	$O(w^2/\alpha)$	✗	good		poor
Iwen's scheme 2	$O(w^2)$	✓	poor		good
Our scheme	$O(w^2/\alpha)$	✓	good		good

Idea: based on disjunct matrix by Porat & Rothschild (2011)

Conclusion and open question

$$\mathcal{M} = \left[\frac{\mathcal{M}_{\mathrm{id}}}{\mathcal{M}_{\mathrm{est}}}\right] \begin{array}{l} \leftarrow \text{subsample from a better } (w,\alpha)\text{-coherent matrix} \\ \leftarrow \text{same} \end{array}$$

(P1): number of measurements	(P2): recovery algorithm runtime
(P3): number of random bits	(P4): error guarantee of $\widehat{\mathbf{x}}$

Lower bounds	$k\log(N/k)$	$k\log(N/k)$	?	ℓ_2/ℓ_2

Paper	**(P1)**	**(P2)**	**(P3)**	**(P4)**
Cormode & Muthukrishnan (2006)	$k\log^3 N$	$k\log^3 N$	$\Omega(N)$	ℓ_2/ℓ_2
Gilbert et al. (2012)	$k\log(N/k)$	$k\log^{\geq 2} N$	$\Omega(N)$	ℓ_2/ℓ_2
Nakos & Song (2019)	$k\log(N/k)$	$k\log^2(N/k)$	$\Omega(N)$	ℓ_2/ℓ_2
Scheme 1, Iwen (2014)	$k\log k\cdot\log N$	$k\log k\cdot\log N$	$\Omega(N)$	ℓ_2/ℓ_1
Scheme 2, Iwen (2014)	$k\log^2 N$	$k\log^2 N$	$\log k\cdot\log(k\log N)$	ℓ_2/ℓ_1
Our scheme	$k\log k\cdot\log N$	$k\log k\cdot\log N$	$\log k\cdot\log(k\log N)$	ℓ_2/ℓ_1

The complexities are subject to O-factor, unless stated with Ω.

$$\boxed{\text{Question: (P1) and (P2) both } O(k\log(N/k))\text{ ? Impossible?}}$$

References

Ba, K. D., Indyk, P., Price, E. & Woodruff, D. P. (2010), 'Lower bounds for sparse recovery', Proceedings of the 2010 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA) pp. 1190–1197.

Cormode, G. & Muthukrishnan, S. (2006), 'Combinatorial Algorithms for Compressed Sensing', International Colloquium on Structural Information and Communication Complexity pp. 280–294.

DeVore, R. A. (2007), 'Deterministic constructions of compressed sensing matrices', Journal of Complexity **23**(4), 918–925.

Gilbert, A. C., Li, Y., Porat, E. & Strauss, M. J. (2012), 'Approximate Sparse Recovery: Optimizing Time and Measurements', SIAM Journal on Computing **41**(2), 436–453.

References (cont.)

Iwen, M. (2014), 'Compressed sensing with sparse binary matrices: Instance optimal error guarantees in near-optimal time', Journal of Complexity **30**(1), 1 – 15.

Johnson, S. (1962), 'A new upper bound for error-correcting codes', IRE Transactions on Information Theory **8**(3), 203–207.

Nakos, V. & Song, Z. (2019), 'Stronger L2/L2 compressed sensing; without iterating', Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing pp. 289–297.

Porat, E. & Rothschild, A. (2011), 'Explicit nonadaptive combinatorial group testing schemes', IEEE Transactions on Information Theory **57**(12), 7982–7989.